Configuration Manual

MSc Internship
MSc Cyber Security

Onyebuchi Aniekwena
Student ID: 15004058

School of Computing
National College of Ireland

Supervisor: Ben Fletcher

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	**Onyebuchi Aniekwena**
Student ID:	**15004058**
Programme:	**MSc in Cyber Security** **Year:** 2020
Module:	**MSc Internship**
Supervisor:	**Ben Fletcher**
Submission Due Date:	**17th August 2020**
Project Title:	**Identification of Key Factors that Influence False Positive Detection & Classification by Anti-Malware Program**
Word Count:	2651 **Page Count** 19

I hereby certify that the information contained in the research work titled "**Identification of Key Factors that Influence False Positive Detection & Classification by Anti-Malware Program"** is information pertaining to the research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.
I agree to an electronic copy of my thesis being made publicly available on NORMA the National College of Ireland's Institutional Repository for consultation.

Signature: ……
17th August 2020
Date: ……

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on a computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Identification of Key Factors that Influence False Positive Detection & Classification by Anti-Malware Program

Configuration Manual Submmission

Onyebuchi Aniekwena
Student ID: x15004058

MSC Internship in Cyber Security

1 Introduction

The configuration manual provides detail of the system set up , hardwar and software specification carried out to ensure seamless implementation of this research Project, **titled "Identification of Key Factors that Influence False Positive Detection & Classification by Anti-Malware Program"**

2 System Configuration

2.1 Hardware

(a) Windows 10, 64-bit. Host Windows Machine a Windows 10, 750 GB Hard Drive
(b) 1T External Drive
(c) AMD E-350 Dual-Core Processor.
(d) Installed Physical Memory (RAM) 8.00GB

2.2 Software

(a) Microsoft Windows 10 Pro, Host version 10.0.18362 Build 18362, System Model VPCEL1E1E, 64-Bit base system type
(b) VirtualBox VM,
(c) Windows 7 VM installed

3 Labset up /VM Setup and Justification:

The motive behind utilising virtualization technology is that it enables us install appropriate virtual machine of my choice, considering the resources available to me in terms of memory capacity other software utilities such as processors available in the host machine. It is also an opportunity to create a sandbox environment where analysis could be conduct seamlessly, without fear of potential malware attack. Setting up of the network for the malware analysis involve choosing VirtualBox VM, with robustness of easily add, delete or duplicate any operating system to aid my analysis.
In order to mitigate easy spread of malware through internet while in attempt to analyse the malware, the host machine was configured to avoid VM escape to secured against the spread of malware, in the virtual environment.

Therefore, the network is configured in such a way that it has limited access to internet in order not to allow it using any channel to spread malicious programs into other machines especially the host machine,while trying to execute suspicious sample.

Due to requirement to work in online registries and resources such as virus total for the analysis that require internet access. The virtual machines was configure with NAT access, as NAT network only allow the virtual machine to initiate the connection, thereby the NAT virtual machine will have the addresses in range, which are inaccessible from the host but only the virtual machines themselves can contact the internet. The virtual machine uses hosts IP address to connect to the machines on the LAN and on the whole internet. Which means other machines on the LAN cannot connect to the virtual machine that is behind NAT, but will ensure connectivity to a public network.

Setting up of the network for the malware analysis involve initial installation of VirtualBox to the personal windows machine, which serve as the host for the network. After which windows operating systems was install in the VirtualBox. For the specific purpose of this project, I install windows 7 VM, IE Version: 11.0.9600.18860, the rational for choosing this version of windows is due to limited memory resources that could impact windows host machine. Oracle VirtualBox, VM equipped with Dynamic Host Configuration Protocol (DHCP)

Static Analysis in VM Sandbox Environment at Online Public registry Virustotal[1]

1

4 INSTALLLATION OF TOOLS

- STEP 1 : Download and Installation of R and Rstudio [2] from https://cran.r-project.org/bin/windows/base/
- ```
  R version 3.6.3 (2020-02-29) -- "Holding the Windsock"
  Copyright (C) 2020 The R Foundation for Statistical Computing
  Platform: x86_64-w64-mingw32/x64 (64-bit)

  R is free software and comes with ABSOLUTELY NO WARRANTY.
  You are welcome to redistribute it under certain conditions.
  Type 'license()' or 'licence()' for distribution details.

    Natural language support but running in an English locale

  R is a collaborative project with many contributors.
  Type 'contributors()' for more information and
  'citation()' on how to cite R or R packages in publications.

  Type 'demo()' for some demos, 'help()' for on-line help, or
  'help.start()' for an HTML browser interface to help.
  Type 'q()' to quit R.
  ```
  - STEP 2: Installation of R packages via Rstudio:

#The first thing to do is to install these packages. We only have to install these packages
#once i.e. we only have to run the function "install.packages" once.
install.packages("e1071")
install.packages('caTools')
install.packages('class')
install.packages('readr')
install.packages('stringr')
install.packages('caret')
install.packages('car')
install.packages('data.table')
install.packages("fBasics")
install.packages("dplyr")

[1] https://www.virustotal.com/gui/home. [2] https://cran.r-project.org/bin/windows/base/
  ```

```
install.packages("DataExplorer")
install.packages("Hmisc")
install.packages("pastecs")
install.packages("wvioplot")
install.packages("doBy")
install.packages("lsr")

install.packages("psych")
install.packages("moments")
```

- STEP:3 Detailing Relevant Libraries

```
# the next step  is to load these packages into the current R session. To do so we can
#use the function "library". Everytime you start a new R session you have to use the library
# command again.

library(e1071)      # for skewness
library(caTools)    #For splitting the dataset into training and test set
library(data.table) # for reading and manipulation of data
library(dplyr)      # for data manipulation and joining
library(ggplot2)    # for ploting
library(caret)      # for modeling
library(xgboost)    # for building XGBoost model
library(caret)      #for modelling
library(car)
library(stringr)
library(readr)
library(class)
```

- STEP 4

```
#The first thing to do is to install these packages. We only have to install these packages
#once i.e. we only have to run the function "install.packages" once.
install.packages("e1071")
install.packages('caTools')
install.packages('class')
install.packages('readr')
install.packages('stringr')
install.packages('caret')
install.packages('car')
install.packages('data.table')
install.packages("fBasics")
install.packages("dplyr")
install.packages("DataExplorer")
install.packages("Hmisc")
install.packages("pastecs")
install.packages("wvioplot")
install.packages("doBy")
install.packages("lsr")

install.packages("psych")
install.packages("moments")
```

AND

```
# the next step  is to load these packages into the current R session. To do so we can
#use the function "library". Everytime you start a new R session you have to use the library
# command again.
```

```
library(e1071)      # for skewness
library(caTools)    #For splitting the dataset into training and test set
library(data.table) # for reading and manipulation of data
library(dplyr)      # for data manipulation and joining
library(ggplot2)    # for ploting
library(caret)      # for modeling
library(xgboost)    # for building XGBoost model
library(caret)      #for modelling
library(car)
library(stringr)
library(readr)
library(class)
```

STEP 4: Setting the working directory

```
setwd("C:/Users/igbou/Desktop/CA MSC CYBER")
```

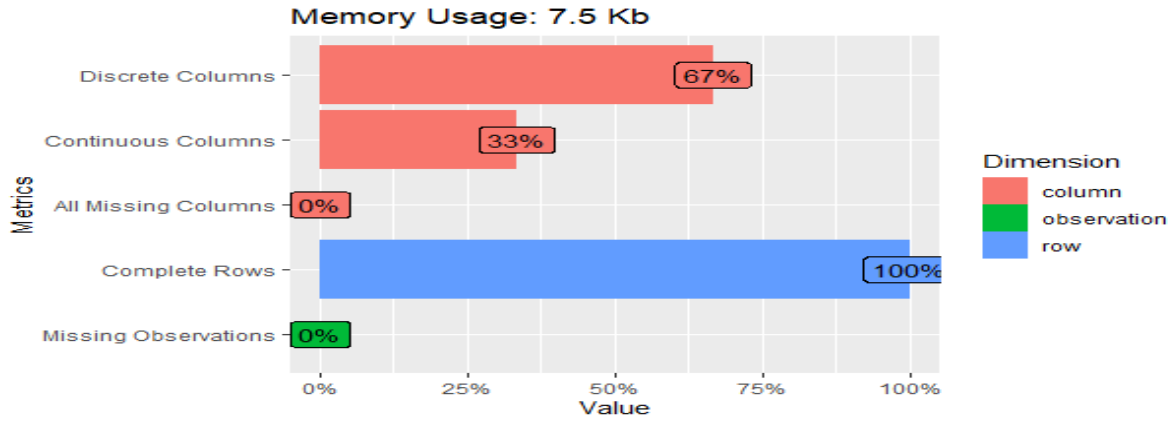
STEP 5: Importing the CSV dataset by reading the data set and assigning it to an object named " dataset".

```
 dataset = read.csv("FP-RULES.csv")
```

STEP 5 Start writing Codes

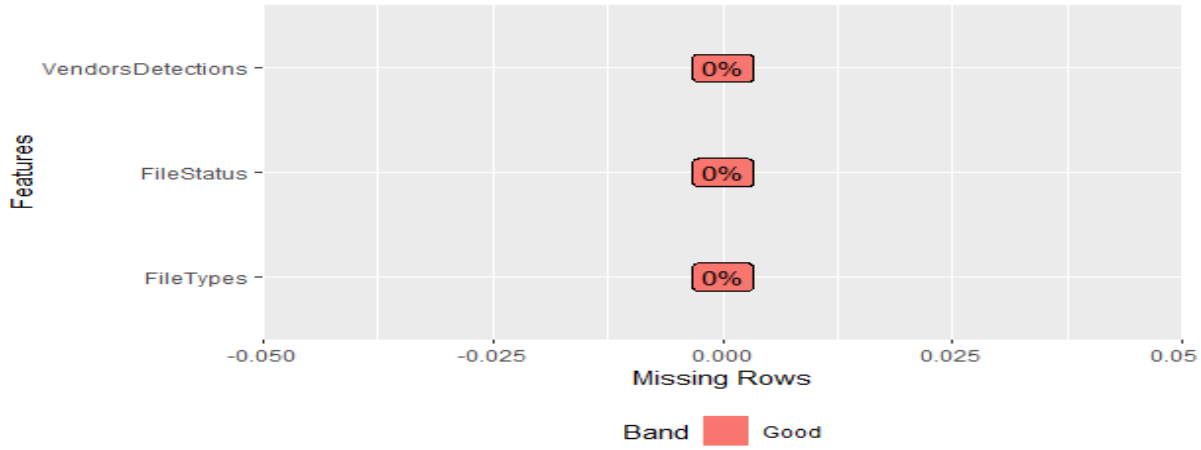
5 Dataset Visualisation from Rstudio to ensure compliance

```
plot_intro( dataset)
```

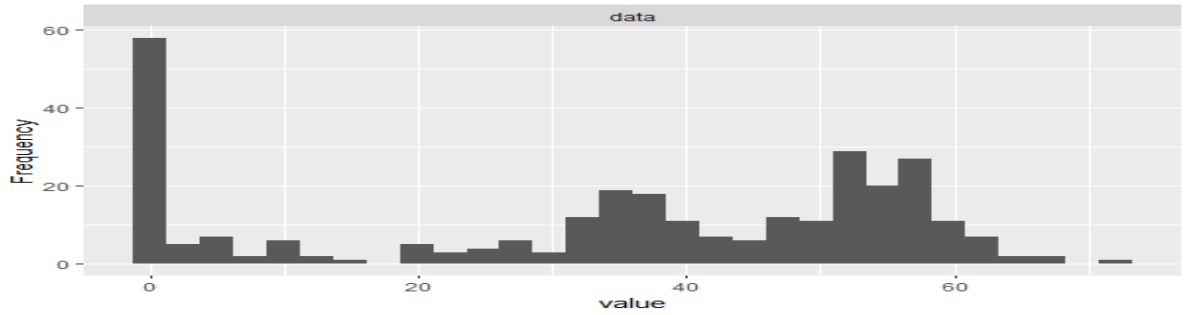



```
plot_str(data =  dataset)
```

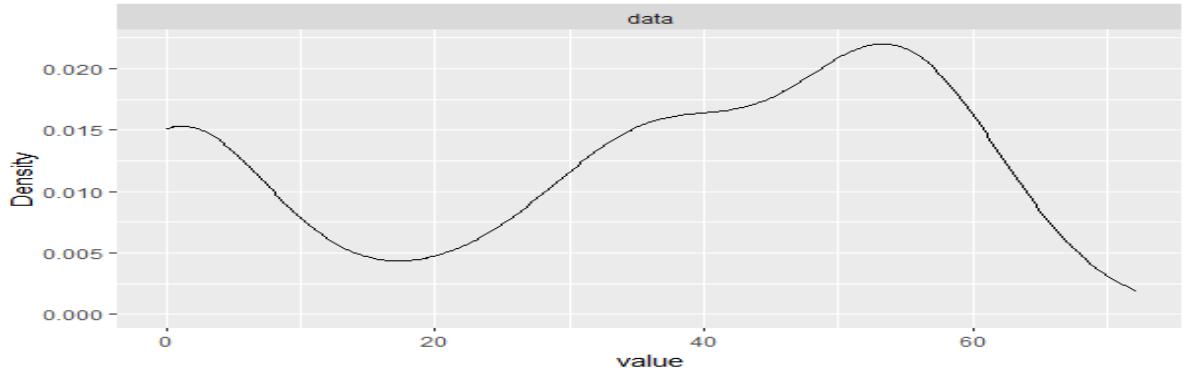
plot_missing(dataset)



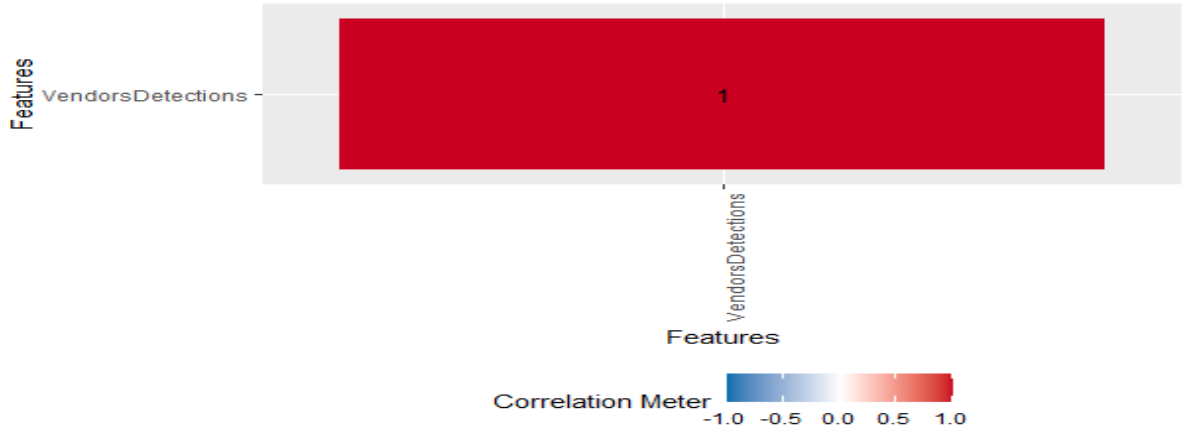
plot_histogram(VendorsDetections)



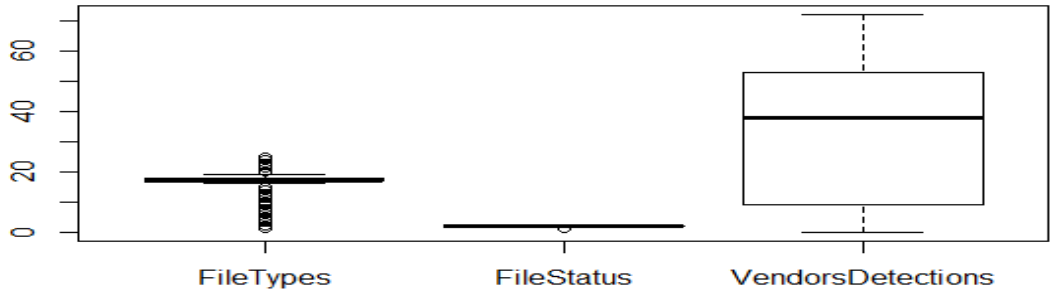
plot_correlation(dataset, type = 'continuous')



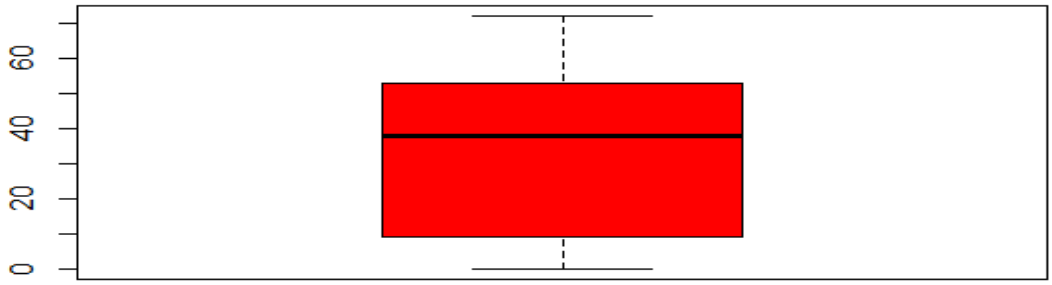
plot_correlation(dataset, type = 'continuous')



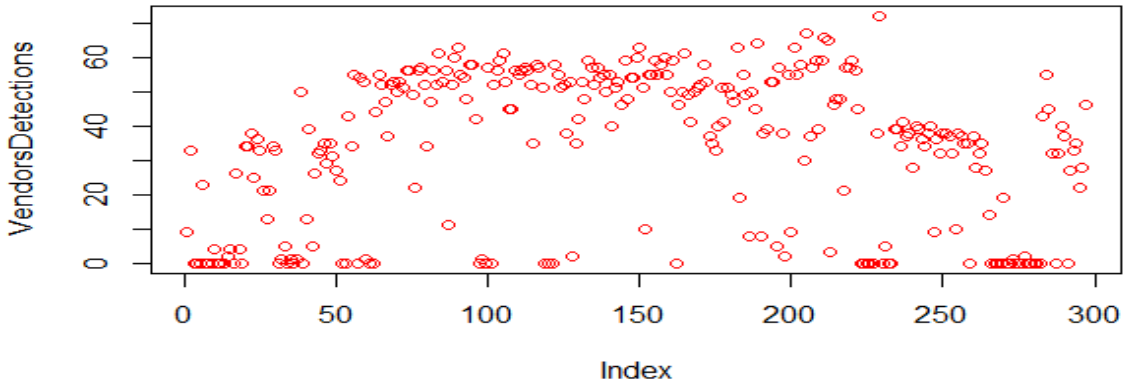
boxplot(dataset,-c(1,2,3,4))

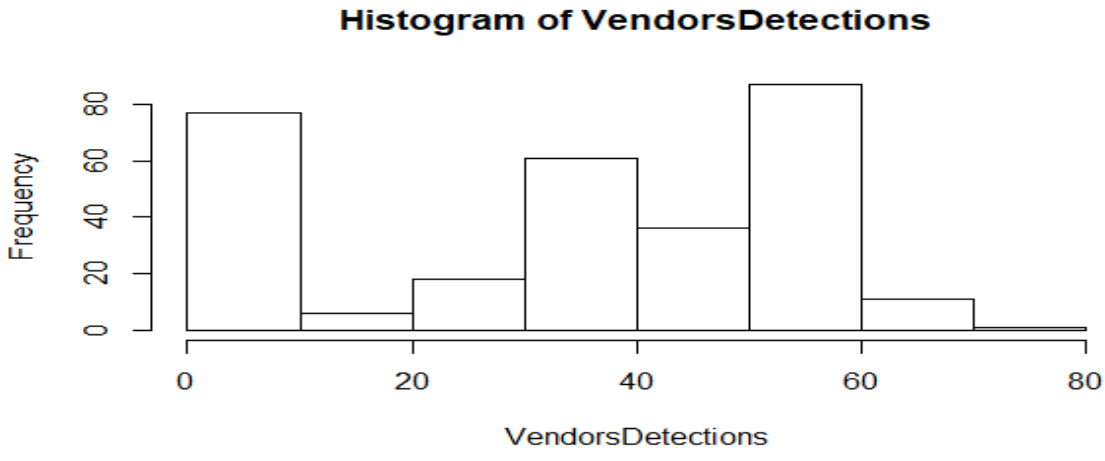


boxplot(VendorsDetections, col = "red")



boxplot(VendorsDetections, col = "red")





File Type Status


```
Console   Terminal   Jobs
C:/Users/igbou/Desktop/CA MSC CYBER/
> library(e1071)       # for skewness
> library(caTools)     #For splitting the dataset into training and test set
> library(data.table) # for reading and manipulation of data
> library(dplyr)       # for data manipulation and joining
> library(ggplot2)     # for ploting
> library(caret)       # for modeling
> library(xgboost)     # for building XGBoost model
> library(caret)       #for modelling
> library(car)
> library(stringr)
> library(readr)
> library(class)
> # setting the working directory
> setwd("C:/Users/igbou/Desktop/CA MSC CYBER")
> #reading the data set and assigning it to an object named " dataset".
>  dataset = read.csv("FP-RULES.csv")
>  dataset =  dataset[,-1]
> #Displaying details of the object " dataset".
> ls.str( dataset)
FileStatus :  Factor w/ 2 levels "FP","TP": 2 2 1 1 1 2 1 1 1 2 ...
FileTypes :  Factor w/ 25 levels "7-zip","ACE",..: 1 2 3 3 3 3 3 3 3 3 ...
VendorsDetections :  int [1:297] 9 33 0 0 0 23 0 0 0 4 ...
> #attaching the  dataset to the R search path.
> attach( dataset)
The following objects are masked from dataset (pos = 3):

    FileStatus, FileTypes, VendorsDetections

> plot_intro( dataset)
> plot_str(data =  dataset)
> plot_missing( dataset)
> plot_intro( dataset)
> plot_str(data =  dataset)
> plot_missing( dataset)
> plot_histogram(VendorsDetections)
> plot_density(VendorsDetections)
> plot_correlation( dataset, type = 'continuous')
> boxplot( dataset,c(1,2,3,4))
```

```
Console   Terminal ×   Jobs ×
C:/Users/igbou/Desktop/CA MSC CYBER/
> boxplot( dataset,-c(1,2,3,4))
> boxplot(VendorsDetections, col = "red")
> class( dataset$FileTypes)
[1] "factor"
> class( dataset$FileStatus)
[1] "factor"
> class( dataset$VendorsDetections)
[1] "integer"
> plot(FileTypes, col = "green") # Frequency (graphically)
> ###### Nominal
> plot(FileStatus, col = "green") # Frequency (graphically)
> table(FileStatus) # Frequency (numerically)
FileStatus
 FP  TP
 64 233
> count( dataset,"FileStatus") # Frequency from plyr package
  FileStatus freq
1         FP   64
2         TP  233
> mean( dataset$FileStatus == "TP") #fraction
[1] 0.7845118
> mean( dataset$FileStatus == "FP") #fraction
[1] 0.2154882
> plot(VendorsDetections, col = "red") # Frequency (graphically)
> table(VendorsDetections) # Frequency (numerically)
VendorsDetections
 0  1  2  3  4  5  8  9 10 11 13 14 19 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35
52  6  4  1  3  4  2  3  2  1  2  1  2  3  2  1  1  1  2  3  3  1  1  1  6  6  7  9
36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 63 64
 3  8 10  6  5  3  2  2  1  5  4  3  5  4  7  8 10 11  5 15 10 10  7  8  3  3  4  1
65 66 67 72
 1  1  1  1
> count(VendorsDetections) # Frequency from plyr package
    x freq
1   0   52
2   1    6
3   2    4
```

```
C:/Users/igbou/Desktop/CA MSC CYBER/
58 66    1
59 67    1
60 72    1
> perc.rank = trunc(rank(VendorsDetections, ties.method = "min")) /
+   length(VendorsDetections) # Percentile rank
> View(cbind(VendorsDetections, perc.rank))
> #Calculate and report the following descriptive statistics for both the FileTypes
> #and FileStatus
> # Mean, Median, Variance, Standard Deviation.
> basicStats(VendorsDetections)
            VendorsDetections
nobs               297.000000
NAs                  0.000000
Minimum              0.000000
Maximum             72.000000
1. Quartile          9.000000
3. Quartile         53.000000
Mean                33.952862
Median              38.000000
Sum              10084.000000
SE Mean              1.274248
LCL Mean            31.445129
UCL Mean            36.460595
Variance           482.241014
Stdev               21.959987
Skewness            -0.468954
Kurtosis            -1.247963
> summary(VendorsDetections)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   0.00    9.00   38.00   33.95   53.00   72.00
> ######################################
> hist(VendorsDetections, breaks = "scott") # histogram using Scott's normal reference
 rule
> quantile(VendorsDetections) # calculating quantiles
  0%  25%  50%  75% 100%
   0    9   38   53   72
> boxplot(VendorsDetections) # boxplot
> quantile(VendorsDetections, c(.05,.95)) # choOsing the 90 percentile
```

```
C:/Users/igbou/Desktop/CA MSC CYBER/
> boxplot(VendorsDetections) # boxplot
> quantile(VendorsDetections, c(.05,.95)) # choOsing the 90 percentile
  5%  95%
 0.0 59.2
> median(VendorsDetections) # calculating the median
[1] 38
> mean(VendorsDetections) # calculating the mean
[1] 33.95286
> sd(VendorsDetections) # calculating the standard deviation
[1] 21.95999
> ###### Ratio Type
> boxplot(VendorsDetections)
> ###### Ratio Type
> boxplot(VendorsDetections)
> boxplot(VendorsDetections[FileStatus == "TP"], VendorsDetections[FileStatus == "FP"],
+         names =  c("TP", "FP"), ylab = "VendorsDetections")
> mean(VendorsDetections[FileStatus == "TP"])
[1] 42.897
> sd(VendorsDetections[FileStatus == "TP"])
[1] 14.84888
> coef.vari.TP = sd(VendorsDetections[FileStatus == "TP"]) /
+   mean(VendorsDetections[FileStatus == "TP"]) * 100 # The Coefficient of Variation fo
r a single variable
> boxplot(VendorsDetections[FileStatus == "FP"], VendorsDetections[FileStatus == "TP"],
+         names =  c("FP", "TP"), ylab = "VendorsDetections")
> mean(VendorsDetections[FileStatus == "FP"])
[1] 1.390625
> sd(VendorsDetections[FileStatus == "FP"])
[1] 8.989619
> coef.vari.FP = sd(VendorsDetections[FileStatus == "FP"]) /
+   mean(VendorsDetections[FileStatus == "FP"]) * 100
> VendorsDetections[2]/VendorsDetections[1]# Element 2 is approx. 39% LOWER than elemen
t 1
[1] 3.666667
> basicStats(VendorsDetections)# This function from the fBasics package will return mos
t of the
```

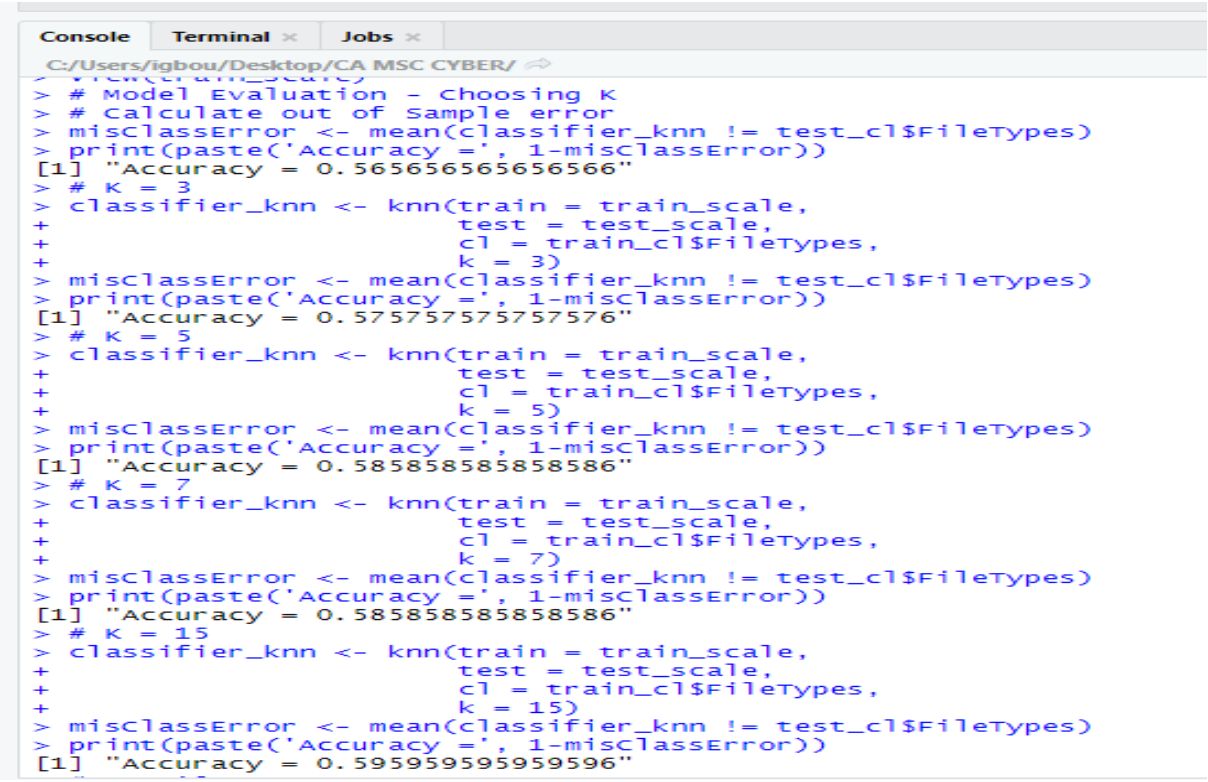
```
 basicStats(VendorsDetections)# This function from the fBasics package will return mos
 of the
            VendorsDetections
obs                297.000000
As                   0.000000
inimum               0.000000
aximum              72.000000
. Quartile           9.000000
. Quartile          53.000000
ean                 33.952862
edian               38.000000
um               10084.000000
E Mean               1.274248
CL Mean             31.445129
CL Mean             36.460595
ariance            482.241014
tdev                21.959987
kewness             -0.468954
urtosis             -1.247963
 basicStats(VendorsDetections[FileStatus == "FP"])
            X..VendorsDetections.FileStatus.....FP.
obs                                      64.000000
As                                        0.000000
inimum                                    0.000000
aximum                                   72.000000
. Quartile                                0.000000
. Quartile                                0.000000
ean                                       1.390625
edian                                     0.000000
um                                       89.000000
E Mean                                    1.123702
CL Mean                                  -0.854915
CL Mean                                   3.636165
ariance                                  80.813244
tdev                                      8.989619
kewness                                   7.568618
urtosis                                  56.471370
```

```
:/Users/igbou/Desktop/CA MSC CYBER/
 # Loading package
 library(e1071)       # for skewness
 library(caTools)  #For splitting the dataset into training and test set
 library(class)
 library(data.table) # for reading and manipulation of data
 library(dplyr)      # for data manipulation and joining
 library(ggplot2)    # for ploting
 library(caret)      # for modeling
 library(xgboost)    # for building XGBoost model
 # setting the working directory
 setwd("C:/Users/igbou/Desktop/CA MSC CYBER")
 #reading the data set and assigning it to an object named " dataset".
 dataset = read.csv("FP-RULES.csv")
 dataset =  dataset[,-1]
 # Splitting data into train
 # and test data
 split = sample.split(dataset, SplitRatio = 0.8)
 train_cl <- subset(dataset, split == "TRUE")
 test_cl <- subset(dataset, split == "FALSE")
 # Feature Scaling
 train_scale <- scale(train_cl[ 3])
 test_scale <- scale(test_cl[ 3])
 # Fitting KNN Model
 # to training dataset
 classifier_knn <- knn(train = train_scale,
                       test = test_scale,
                       cl = train_cl$FileTypes,
                       k = 1)
 classifier_knn
#
```

```
                    K = 1)
> classifier_knn
 [1] PE32      XML       XML       ASCII     XML       XML       XML       PE32
 [9] RAR       Composite XML       XML       data      Composite Composite CDFV2
[17] PE32      XML       ACE       PE32      XML       PE32      RAR       PE32
[25] PE32      Composite PE32      PE32      PE32      PE32      PE32      PE32
[33] XML       PE32      PE32      PE32      PE32      PE32      CDFV2     PE32
[41] XML       PE32      PE32      RAR       PE32      PE32      PE32      PE32
[49] PE32      PE32      PE32      PE32      PE32      PE32      PE32      PE32
[57] PE32      PE32      Zip       PE32      Zip       PE32      PE32      PE32
[65] PE32      PE32      PE32      PE32      PE32      PE32      PE32      PE32
[73] Composite PE32      XML       XML       PE32      XML       RAR       RAR
[81] PE32      Composite PE32      PE32      RAR       RAR       XML       ISO
[89] Composite XML       XML       XML       PE32      XML       PE32      ISO
[97] PE32      ASCII     Composite
25 Levels: 7-zip ACE ASCII CDFV2 Composite data DOS ELF gzip ISO Java ... Zip
```

```
> # Confusiin Matrix
> cm <- table(test_cl$FileTypes, classifier_knn)
> cm
           classifier_knn
            7-zip ACE ASCII CDFV2 Composite data DOS ELF gzip ISO Java JPEG
  7-zip         0   0     0     0         0    0   0   0    0   0    0    0
  ACE           0   0     0     0         0    0   0   0    0   0    0    0
  ASCII         0   0     1     0         0    0   0   0    0   0    0    0
  CDFV2         0   0     0     0         0    0   0   0    0   0    0    0
  Composite     0   0     0     0         1    0   0   0    0   0    0    0
  data          0   0     0     0         0    1   0   0    0   0    0    0
  DOS           0   0     0     0         0    0   0   0    0   0    0    0
  ELF           0   0     0     0         1    0   0   0    0   0    0    0
  gzip          0   0     0     0         0    0   0   0    0   0    0    0
  ISO           0   0     0     1         1    0   0   0    0   0    0    0
  Java          0   0     0     0         0    0   0   0    0   0    0    0
  JPEG          0   0     0     0         0    0   0   0    0   0    0    0
  Microsoft     0   0     0     0         0    0   0   0    0   0    0    0
  MS            0   1     0     0         0    0   0   0    0   0    0    0
  MS-DOS        0   0     0     0         0    0   0   0    0   0    0    0
  PDF           0   0     0     0         0    0   0   0    0   0    0    0
  PE32          0   0     0     1         2    0   0   0    0   0    0    0
  PE32+         0   0     0     0         0    0   0   0    0   0    0    0
  PHP           0   0     0     0         0    0   0   0    0   0    0    0
  RAR           0   0     0     0         1    0   0   0    0   0    0    0
  Rich          0   0     0     0         0    0   0   0    0   0    0    0
  UDF           0   0     0     0         0    0   0   0    0   1    0    0
  UTF-8         0   0     0     0         1    0   0   0    0   0    0    0
  XML           0   0     0     0         0    0   0   0    0   0    0    0
  Zip           0   0     1     0         1    0   0   0    0   1    0    0
```

```
           classifier_knn
            Microsoft MS MS-DOS PDF PE32 PE32+ PHP RAR Rich UDF UTF-8 XML Zip
  7-zip             0  0      0   0    1     0   0   0    0   0     0   0   0
  ACE               0  0      0   0    0     0   0   0    0   0     0   0   0
  ASCII             0  0      0   0    0     0   0   0    0   0     0   5   0
  CDFV2             0  0      0   0    0     0   0   0    0   0     0   0   0
  Composite         0  0      0   0    1     0   0   1    0   0     0   0   0
  data              0  0      0   0    0     0   0   0    0   0     0   2   0
  DOS               0  0      0   0    0     0   0   0    0   0     0   0   0
  ELF               0  0      0   0    0     0   0   0    0   0     0   0   0
  gzip              0  0      0   0    0     0   0   0    0   0     0   0   0
  ISO               0  0      0   0    1     0   0   0    0   0     0   0   0
  Java              0  0      0   0    0     0   0   0    0   0     0   0   0
  JPEG              0  0      0   0    0     0   0   0    0   0     0   1   0
  Microsoft         0  0      0   0    0     0   0   0    0   0     0   0   0
  MS                0  0      0   0    0     0   0   0    0   0     0   0   0
  MS-DOS            0  0      0   0    0     0   0   0    0   0     0   0   0
  PDF               0  0      0   0    0     0   0   0    0   0     0   0   0
  PE32              0  0      0   0   45     0   0   2    0   0     0   3   2
  PE32+             0  0      0   0    1     0   0   0    0   0     0   2   0
  PHP               0  0      0   0    0     0   0   0    0   0     0   1   0
  RAR               0  0      0   0    3     0   0   4    0   0     0   0   0
  Rich              0  0      0   0    0     0   0   0    0   0     0   1   0
  UDF               0  0      0   0    0     0   0   0    0   0     0   0   0
  UTF-8             0  0      0   0    0     0   0   0    0   0     0   0   0
  XML               0  0      0   0    1     0   0   0    0   0     0   4   0
  Zip               0  0      0   0    2     0   0   0    0   0     0   0   0
```



6 References

[1] Q. Luo, “Advancing knowledge discovery and data mining” in *Proceedings - 1st International Workshop on Knowledge Discovery and Data Mining, WKDD* 3–5 (2008). doi:10.1109/WKDD.2008.153